IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of

Ditlow, et al.

Serial No.: 09/943,829 Group Art Unit: 2195

Filed: August 31, 2001 Examiner: Tang, K.

For: **METHOD AND APPARATUS TO MANAGE MULTI-COMPUTER SUPPLY**

Commissioner of Patents
Alexanderia, VA 22313-1450

APPELLANTS' SUPPLEMENTAL BRIEF ON APPEAL

Sir:

Appellants respectfully appeal the rejection of claims 1-20 in the Office Action dated April 19, 2005. A Notice of Appeal was timely filed on July 19, 2005, and a Brief on Appeal was timely filed on September 19, 2005.

In an Office Action mailed on November 29, 2005, the Examiner re-opened prosecution, citing new references, US Patent Application Publication US 2002/0059625 to Kurauchi, previously-cited US Patent 6,105,053 to Kimmel et al., and newly-cited US Patent 6,016,503 to Overby et al. Appellants have carefully evaluated this new rejection and believe that it fails to further prosecution in any meaningful manner and have concurrently filed a Petition Under 37 CFR §1.181 to Re-instate Appeal.

This Supplemental Brief is submitted in response to this latest Office Action, explaining how the new rejection in this Office Action is no better than the rejection currently in the process of Appeal.

This Supplemental Brief contains only a discussion of the deficiencies of the new rejection and does not repeat the formalities and information of the Appeal Brief filed on September 19, 2005.

STATUS OF CLAIMS

Claims 1-20, all of the claims presently pending in the application, stand rejected on prior art grounds.

More specifically, claims 1-4, 6-10, 12-16, and 18-20 stand rejected under 35 USC §103(a) as unpatentable over US Patent Application US 2002/0059625 to Kurauchi, further in view of US Patent 6,105,053 to Kimmel et al., and claims 5, 11, and 17 stand rejected under 35 USC §103(a) as unpatentable over Kurauchi/Kimmel, further in view of US Patent 6,016,503 to Oberby et al.

SUMMARY OF CLAIMED SUBJECT MATTER

Appellants' invention, as disclosed and claimed in independent claim 1, is directed to a computer-implemented method for determining a listing of hosts on a network to perform a parallel application (lines 15-16 of page 1, lines 16-19 of page 4, lines 1-3 of page 6, Figure 1), including determining a listing of all possible hosts on the network for performing the parallel application (lines 16-18 of page 1, line 19 of page 6 through line 1 of page 7). For each of the possible hosts (line 2 of page 18) a current capacity and a current utilization is determined and a difference between the current capacity and the current utilization is calculated (lines 2-4 of page 7, line12 of page 13, line 3 of page 18 through line 9 of page 19, equation [4] in Figure 7). A listing of hosts is selected from the listing of all possible hosts, based on sorting the calculated differences (lines 18-19 of page 1, lines 4-5 of page 7, lines 6-21 of page 24, Equation [6a] of Figure 7).

THE NEW PRIOR ART REJECTIONS BASED ON KURAUCHI

The Examiner alleges that the present invention defined by claims 1-4, 6-10, 12-16, and 18-20 are rendered obvious when newly-cited primary reference, US Patent Application Publication US 2002/0059625 to Kurauchi, is modified by previously-cited

secondary reference, US Patent 6,105,053 to Kimmel et al., and that claims 5, 11, and 17 are rendered obvious when Kurauchi/Kimmel is further modified by newly-cited US Patent 6,016,503 to Overby et al.

In summary, after careful review, Appellants submit that the newly-cited primary reference Kurauchi is non-analogous art relative to the present invention, since it does not address a parallel processing environment as that term of art is understood by one having ordinary skill in the art, and cannot be modified into the parallel-processing environment defined by the claimed invention without changing its principle of operation. Moreover, Appellants submit that Kimmel is non-analogous to this new primary reference and, therefore, not properly combinable and that, even if primary reference Kurauchi were to be modified in accordance with Kimmel, the combination of references would not result in the claimed invention defined in claims 1-4, 6-10, 12-16, and 18-20.

The rejection for claims 5, 11, and 17 is somewhat confused, since it refers to Robertazzi, so it is discounted for this response.

APPELLANTS' ANALYSIS OF THE REJECTION BASED ON KURAUCHI

Appellants first point out that primary reference Kurauchi is not addressing the parallel processing environment as required by the independent claims. That is, as well understood by one having ordinary skill in the art, parallel processing involves, in general, the distribution of processing tasks to be executed independently on separate CPUs. This generalized description is confirmed by the following definition from the 6th Edition of the "Dictionary of Computer and Internet Terms", D. A. Downing, Ph.D., et al., 1998, Barron's Educational Series, Inc.:

"PARALLEL PROCESSING computation carried out at the same time on different CPUs, or on a CPU that can execute more than one instruction at the exact same time. By contrast, most multitasking is accomplished by making a single CPU switch its attention among several tasks. This is called concurrent processing or timesharing."

Appellants submit that the processing described in primary reference Kurauchi is multitasking, not parallel processing that is addressed by the present invention. Simply

using the word "parallel" or "in parallel" does not necessarily bestow the property of "parallel processing", even if a CPU is involved.

The Examiner points to paragraphs [0048] and [0049] in Kurauchi as satisfying the description in the preamble of claim 1 that this reference addresses "a parallel application with host processors on a network." However, these two paragraphs merely use the word "executed in parallel" as describing parallel operations executed in parallel by designated different hardware modules within nonlinear editing device 1101, under control of a single CPU 2111.

This simple single-CPU architecture demonstrates the multitasking operation described above and fails to have the characteristics of parallel processing. That is, in order to convert the nonlinear editing device 1101 of Kurauchi into a parallel processing environment, would require, first, that the problem being computed be of the nature that it can be broken down into separate tasks and distributed to different CPUs which then perform their respective calculation tasks in parallel and return the respective results to a central location to be merged and/or redistributed for additional processing.

The simple broadcast system of Kurauchi does not address this type of parallel processing problem. Rather, in Kurauchi each hardware module performs its respective and independent task, designed as an independent specific broadcasting task assigned by the CPU. When its task is completed, the hardware modules of Kurauchi have no processing result that is a partial computation of a parallel processing problem to be returned to a central location to be merged into results from the other modules.

Therefore, the task assignment process in this reference is merely that of ensuring that the various independent tasks related to broadcasting are appropriately assigned to an available hardware module within the limits of the nonlinear editing device's resources. This assignment process is not in any way related to assigning tasks for a parallel processing problem, as this term is understood in the art, even if the broadcast tasks are separately being executed in parallel by the various hardware modules.

Therefore, Appellants submit that primary reference Kurauchi is non-analogous art to that of the present invention that addresses an assignment of tasks for a parallel processing problem.

Moreover, in order to satisfy the requirement described by the independent claims, the function of the nonlinear editing device 1101 would have to be changed into one that fits the description of parallel processing. Such change in function would clearly change the principle of operation of this primary reference Kurauchi and is, therefore, forbidden by the holding in *In re Ratti,* 270 F.2d 810, 123 USPQ 349 (CCPA, 1959), as described in MPEP §2143.02:

"If the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims prima facie *obvious."*

Hence, turning to the clear language of the claims, in Kurauchi there is no teaching or suggestion of: ".... determining a listing of hosts on a network to perform a parallel application, said method comprising: determining a listing of all possible hosts on said network for performing said parallel application ....", as required by independent claim 1. The remaining independent claims also make reference to the parallel processing environment.

Moreover, Appellants submit that there are other deficiencies inherent in Kurauchi that the new rejection simply ignores.

First, independent claim 1 clearly requires that there be a number of hosts on a network that can potentially participate in the parallel processing. Although Figure 2 of Kurauchi shows an external network, it has a single CPU 2111 of any interest that is discussed. It clearly fails to address more than one CPU interconnected by a network, as required by independent claim1.

Second, the Examiner points to paragraph [0093] of Kurauchi as satisfying the second and third claim limitations related to calculation of the difference of current capacity and current utilization of each host. However, this paragraph clearly describes simply the process of checking whether a resource is currently available, as opposed to currently unavailable because it is currently busy executing its task. This simple availability fails to satisfy the plain meaning of the claim language that the host's current capacity be subtracted from its current utilization.

Even the description that "... the hardware resource management unit 3031 calculates an available amount of the hardware resource, by subtracting a presently occupied amount of the hardware resource from a total amount of the hardware resource" is a reference to determining how many of the redundant units available for performing that task are currently available for service. This is an entirely different concept from that required by the plain meaning of the claim language, if each hardware resource were to be a host available for participating in a parallel processing problem, that the current capacity be subtracted from the current utilization for each potential host.

Third, because there is no calculation in Kurauchi of a difference between current capacity and current utilization, there clearly is no sorting of this calculation. The processing in paragraphs [0094], [0173], and [0192], to which the Examiner points are merely describing that the redundant/alternate-function hardware modules are selected, based upon current availability and the priorities of an editing list, such that the total capacity of the system is not exceeded.

All of these selection/prioritizing concepts are entirely different from that described by the plain meaning of the claim language, wherein it is required that, for each potential host, there be a calculation of the difference between current capacity and current utilization, and the listing of such difference calculations be sorted.

Fourth, the Examiner concedes that Kurauchi fails to suggest using a listing of all possible hosts on the network and invokes secondary reference Kimmel to overcome this deficiency. Appellants submit that there are several problems in this combination.

- These two references are non-analogous, since they address two entirely different problems, and, therefore, cannot be combined in an obviousness rejection (see MPEP §2141.01(a)).
- Converting the simple single CPU system of Kurauchi into a NUMA system completely changes it principle of operation, which conversion is precluded by MPEP §2143.01, as described above.
- Even if Kimmel were to be combined with Kurauchi, it would still fail to overcome the deficiencies identified above for Kurauchi, and for the following reason described in the following paragraphs.

Fifth, Appellants submit that, to one having ordinary skill in the art, Kimmel shows a multiprocessing system (e.g., one machine with several processors) and models this multiprocessor system and its shared resources with a hierarchical tree, in order to efficiently manage the system-shared resources between job processors working on the same task. In contrast, the exemplary embodiment of the present invention provides for a listing of all available hosts available on a network, whether multiprocessor machines or single-CPU machines. It does not include shared-system resources. Indeed, the present invention exemplarily assumes that all processors are independent, the most general type of parallel processing.

Typically, in a parallel processing environment, there is one master and the parallelism is figured out by the master. A cooperative effort to figure out the structure of the parallel application is not trivial and not typical. Both Kimmel and Kurauchi teach environments that are multiprocessor systems (e.g., one machine), where threads or processess are dispatched. This implies some sort of controller in the operating system, rather than a cooperative parallel processing environment.

Kurauchi, in paragraphs [0048] and [0049] is describing sharing the network bandwidth between processes. Kimmel is optimizing assignment of thread groups to job processors by categorizing thread groups by "time-sharing allowable" or relationship to certain data and improves the response time for real-time threads. Although both patents describe events occurring in real-time, neither discusses determining a list of processors, based on historical and current data (e.g., for processor utilization and capacity) for a parallel application.

The Examiner's reference to paragraph [0066] of Kurauchi support the statement that there is an operating system, not the generation of a list of hosts to use for processing an application in parallel, based on capacity and utilization of each of the machines.

Kimmel determines composite load (e.g., work to be done) as the weighted, normalized sum of other loads (job processor, memory, and priority loads) but it does not compute the difference between current capacity and current utilization and normalizing that quantity. In contrast, the present invention normalizes a "supply" of hosts to help determine where to place the work.

Kurauchi is determining which hardware to secure for transferring data and determines if network bandwidth can be used for several transfers that are processed by different units. In contrast, the present invention uses loosely- or tightly-coupled CPUs to process an application in parallel.

In Kurauchi, each unit is a special purpose item. In the present invention, any job can run on any host provided it "fits". The criterion is normalized capacity – utilization. The metrics of exemplary embodiment of the present invention for utilization and capacity are: memory, file system, and cache. Kurauchi's measure for "in use" is whether the hardware resource has been allocated to a transfer.

Kurauchi's resource management unit allocates a maximum amount of hardware resource to non-reproduction processing tasks. Whatever is left over is allocated with priority to production jobs. In contrast, in the exemplary embodiment of the present invention, a list is created of the smallest number of hosts to use to solve a parallel application.

Kimmel cannot be extended to include all possible hosts on a network because all hosts may not have shared resources. Kimmel claims an operating system for a multiprocessor system which obviously manages shared resources (memory, paging space, thread contexts) between the processors in the machine. Data affinity may not be a meaningful metric across loosely-coupled processors.

Finally, relative to newly-cited Overby, in the exemplary embodiment of the present invention, the utilization is based on CPU, memory, temporary files, and cache for each host machine and how these properties vary over time. Utilization is combined with capacity, normalized and used with a priority when selecting hosts that will solve a parallel problem.

In contrast, Overby's utilization applies to a shared resource, for example, the amount of shared memory used and this utilization is used to manage the shared resource. This is an entirely different concept from that of managing resources on separate machines, let alone doing so for a parallel processing environment.

Therefore, Appellants submit that Overby adds nothing to overcome the basic deficiencies of Kurauchi and/or Kimmel.

CONCLUSION

In view of the foregoing, Appellants submit that claims 1-20, all the claims presently pending in the application, are clearly enabled and patentably distinct from the prior art of record and in condition for allowance. Thus, the Board is respectfully requested to remove all rejections of claims 1-20, as based on the rejection currently the subject of Appellants Appeal Brief filed on September 19, 2005, and as based on the above-discussed rejection based on newly-cited Kurauchi and Overby.

Please charge any deficiencies and/or credit any overpayments necessary to enter this paper to Assignee's Deposit Account number 09-0456.

Respectfully submitted,

Dated: 2/28/06

Frederick E. Cooperrider
Reg. No. 36,769

McGinn Intellectual Property Law Group, PLLC.
8231 Old Courthouse Road, Suite 200
Vienna, VA 22182-3817
(703) 761-4100
Customer Number: 21254

CERTIFICATION OF TRANSMISSION

I certify that I transmitted via facsimile to (571) 273-8300/-3772 this Supplemental Appeal Brief to Examiner K. Tang on February 28, 2006.

Frederick E. Cooperrider
Reg. No. 36,769